Introduction to Information Theory

Lecture 12 Part C

Lecturer: Haim Permuter *Scribe: Omer Pilosof, Itzik Waizman*

I. REMINDER

In our previous lectures, we delved into channel coding using polar codes. We learned that given a channel $W_{Y|X}$, we can transform this channel into virtual channels and polarize the error probabilities across these channels. By selectively utilizing the reliable virtual channels, we observed that polar codes achieve near-optimal performance in terms of error correction capability and channel capacity. We explored how, given n bits $u_1, \ldots, u_n$, we can use the matrix $G^{\otimes n}$ to encode our bits before transmitting them through the channel, where $G^{\otimes n}$ represents the n-th Kronecker power of the polarizing matrix

$$G^2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}.$$

After transmitting the encoded bits $\{x_1, \ldots, x_n\}$ through the channels, we calculate the Log Likelihood Ratios (LLRs), LLR(x_i), using the received signals $\{y_1, \ldots, y_n\}$, where $\text{LLR}(x_i) = \log\left(\frac{P(x_i=1|y)}{P(x_i=0|y)}\right)$. These LLRs are then used for bit decision, providing us with an estimation of the original bits $u_1, \ldots, u_n$. The process is illustrated in Figure 1.

We also saw that the capacity of the i-th channel is given by $C_i = I(u_i; y^n|u^{i-1})$. Here, k bits of information are sent via reliable channels, where the capacity approaches 1, while $n-k$ frozen bits are sent through noisy channels, where the capacity approaches 0. This gives us a rate of $\frac{k}{n}$, where in the context of channel coding, our goal was to achieve the highest possible rate, constrained by the capacity of the channel $W_{Y|X}$. In the upcoming section, we will explore another significant application of polar codes, specifically data compression.

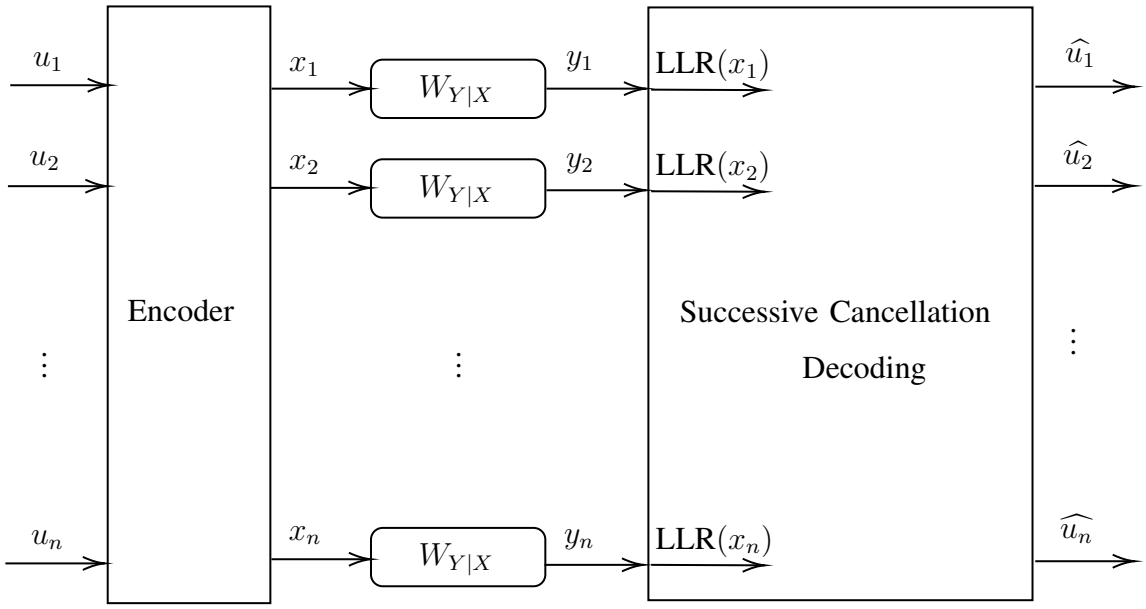


Fig. 1: Polarization process for channel coding.

II. Background

In this lecture, we will explore the application of polar codes for data compression, transitioning from their conventional use in error correction to utilizing the polarization matrix for compression. Polar codes play a crucial role in communication by enabling efficient data compression for transmission over networks. This is particularly beneficial in distributed compression scenarios, such as wireless sensor networks and distributed video coding. As near-capacity channel codes, polar codes exhibit significant potential and have been shown to achieve compression rates close to theoretical limits. This lecture is based on [1].

We begin by outlining our objective. Consider n random variables $Z_1, \ldots, Z_n$, which are independently and identically distributed (i.i.d.) following a Bernoulli distribution, Ber(p). Our aim is to design an encoder-decoder system that reduces the number of transmitted bits from n to a smaller portion k, while still allowing the decoder to accurately reconstruct the original input. Specifically, the estimated values from the decoder, $\hat{Z}_1, \ldots, \hat{Z}_n$, should satisfy the condition:

$$P(\hat{Z}^n \neq Z^n) \leq \varepsilon_n,$$

where $\varepsilon_n \to 0$ as $n \to \infty$. In this context, the compression rate is defined as $\frac{k}{n}$, with k being the number of bits actually transmitted. Unlike channel coding, where the goal is to maximize the rate, in data compression we aim to minimize the rate to achieve efficient data reduction.

III. Compression using polar code

The objective outlined in the previous section can be addressed using the concept of polarization. Given n bits, $z_1, \ldots, z_n$, generated from our n random variables $Z_1, \ldots, Z_n$, we will employ a polarization matrix, as we have done in channel coding, denoted by $G^{\otimes n}$, to achieve the encoded input $w^n = z^n G^{\otimes n}$. This polarization process results in $n - k$ bits that are considered predictable where a bit w_i is deemed predictable if the following condition holds:

$$H(w_i|w^{i-1}) \leq \varepsilon,$$

for a predefined acceptable ε. On the other hand, there are k unpredictable bits, where a bit w_i is deemed unpredictable if $H(w_i|w^{i-1}) \geq 1-\varepsilon$. The k unpredictable bits correspond to k transformed random variables whose distribution approaches Ber$(1/2)$. Initially, all the distributions were the same, Ber(p), but through the transformation, they are polarized. Since the predictable bits can be estimated with high probability using the k unpredictable bits, we can reduce the transmitted data to these k bits, effectively increasing throughput. The decoder will utilize the received signals to reconstruct the original bits. For the bits that were not transmitted, it estimates a sample using the received signals. Specifically, if w_j is a predictable bit and was not transmitted, the decoder will estimate $\hat{w_j}$ using:

$$\hat{w_j} = \arg\max_b P(w_j = b|w^{j-1}).$$

The estimated $\hat{w_i}$ values will then be used to decode the received information and reconstruct the original input data, as we will explain in detail later on. The described

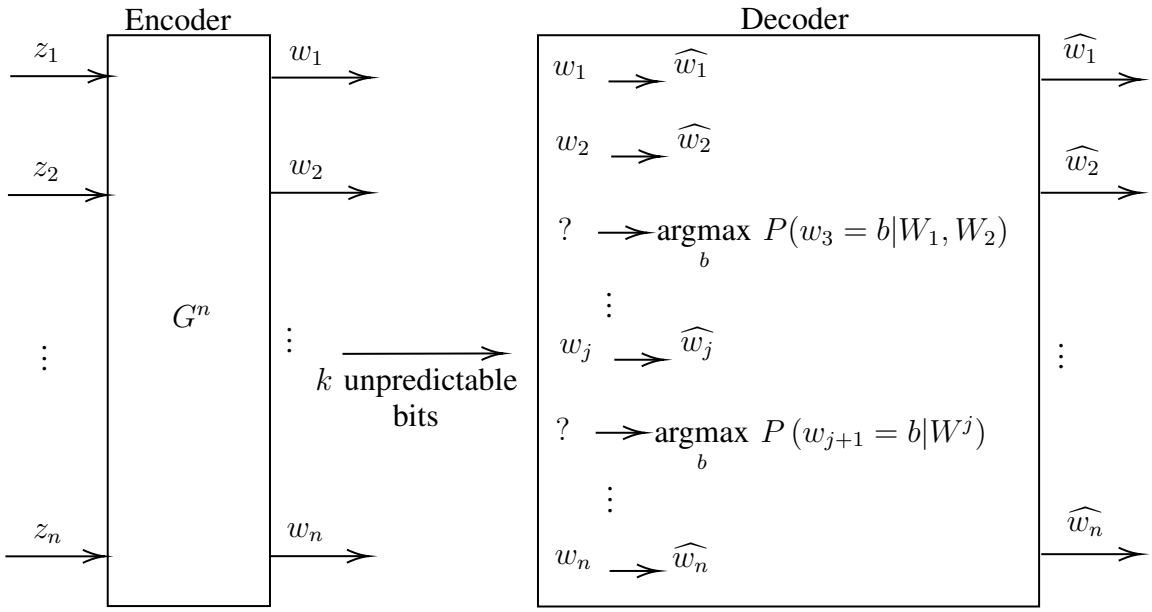
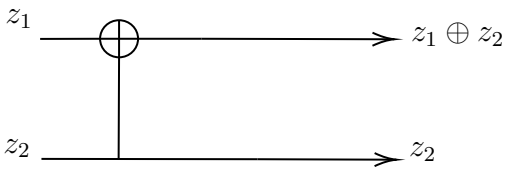


Fig. 2: Polarization process for data compression.

process is illustrated in Figure 2, where $\hat{w}_i$ represents the received signals by the decoder. The '?' symbol denotes a predictable bit that was not transmitted and is estimated by the decoder using the argmax method explained previously.

Example 1 (Polarized Encoding with $n = 2$) Consider two independent and identically distributed Ber(p) random variables, Z_1 and Z_2. The encoding scheme is defined as follows:



Assume $0 \leq p \leq \frac{1}{2}$ for simplification and let $w_1 = z_1 \oplus z_2$, $w_2 = z_2$. Note that

$$P(w_1 = 1) = P(z_1 = 0, z_2 = 1) + P(z_1 = 1, z_2 = 0),$$

and thus

$$P(w_1 = 1) = 2p(1 - p) \geq p.$$

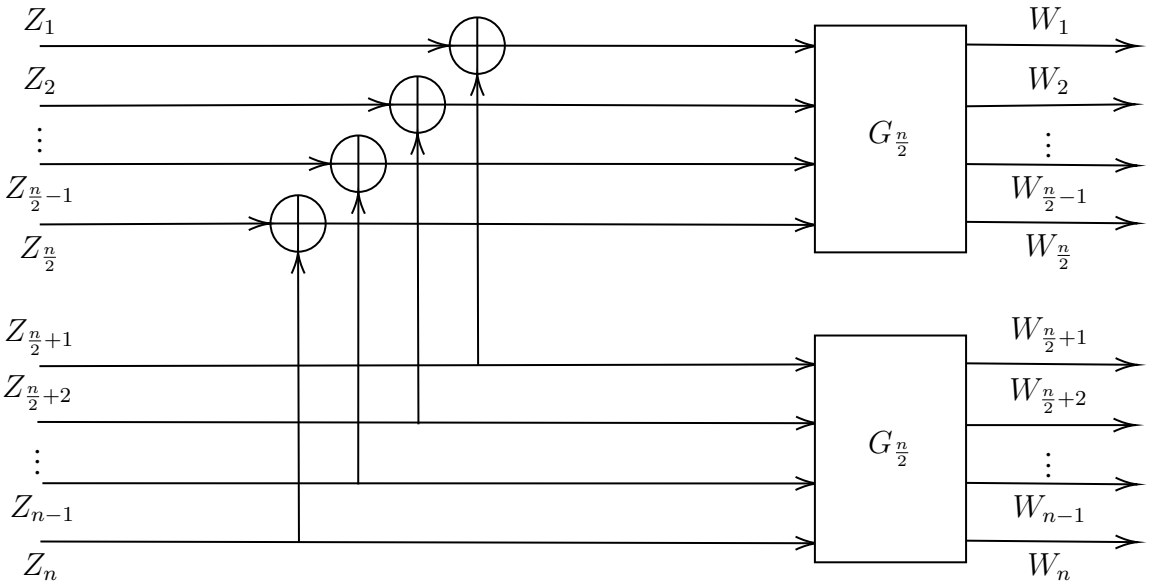


Fig. 3: Polarization scheme for n inputs.

Furthermore, since $(1-2p)^2 \geq 0$, it follows that $2p(1-p) \leq \frac{1}{2}$. Consequently, the entropy of W_1 is increased, i.e., $H(W_1) \geq H(Z_1)$. The transformed distribution remains Bernoulli but moves closer to Ber $\left(\frac{1}{2}\right)$ for larger values of p, while still being less than $\frac{1}{2}$. For W_2, given the injective mapping between W_2 and Z_2, we have $H(W_2) = H(Z_2)$. Moreover, since conditioning reduces entropy, it holds that $H(W_2|W_1) \leq H(Z_2)$. In summary:

$$H(W_1) \geq H(p)$$
$$H(W_2) \leq H(p)$$

The distribution of W_1 approaches Ber $\left(\frac{1}{2}\right)$, identifying W_1 as the unpredictable bit to be transmitted.

Example 2 (Polarized compression with n input) Let $G_{2^n} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes n}$ and $z^n = \{u, v\}$ where $u = \{Z_1, ..., Z_{\frac{n}{2}}\}$ and $v = \{Z_{\frac{n}{2}+1}, ..., Z_n\}$. The polarization scheme is illustrated in Figure 3, where $G_n(z^n) = G_n(u, v) = (G_{\frac{n}{2}}(u), G_{\frac{n}{2}}(v))$. This example demonstrates how polarized compression can be applied to n input bits. This example

shows how a larger polarization matrix G_2^n can be constructed by taking the Kronecker power of the 2x2 polarization matrix G_2. The input vector Z is then partitioned into two subvectors u and v, and the polarization matrix is applied to each subvector separately. The resulting polarized vector W has some bits that are more entropic and some bits that are less entropic. This example illustrates how the concept of polarization can be extended to larger input sizes and how it can be used to achieve efficient compression of data.

We have illustrated how the entropy characteristics of transformed variables differ in the polarization process. Next, we will delve into an analysis of the polarization computational complexity.

A. Polarization Methods

Before proceeding to the computational complexity analysis, we will first recall the polarization methods we have previously learned. Following this review, we will introduce an additional method to further enhance our understanding and simplify the complexity computation.

1) Matrix multiplication

Matrix multiplication is a straightforward method where we simply multiply the input bits with the polarization matrix. Given n bits, the polarization matrix is $G^{\otimes n}$, as presented at the beginning of this lecture. Thus, we have

$$w^n = z^n G^{\otimes n}.$$

2) Diagram Calculation

In this method we calculate the polarization using the diagram described in Figure 4.

3) Tree Method

In this method, we perform recursive vector additions over pairs of nodes that share a common parent in the following way: For each pair of nodes u and v with a common parent, we create a new vector for the parent node defined as $[u+v, v]$. An example with $n = 4$ is given below in Figure 5.

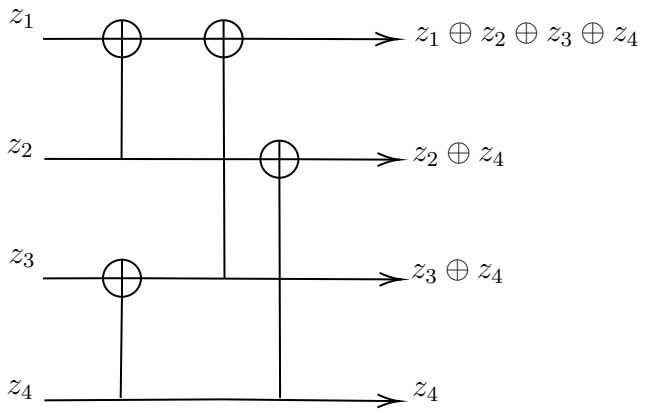


Fig. 4: Polarization calculation using scheme for $n = 4$.

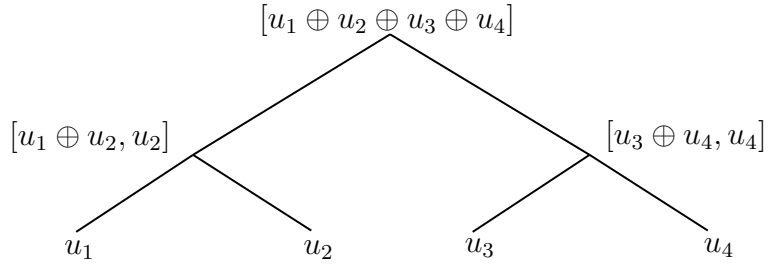


Fig. 5: Polarization calculation using tree diagram.

4) Recursive Formula

Now, we present a new recursive approach to calculate the polarization. This method provides a convenient formulation for complexity analysis. Let $z = [u, v]$ where $z \in \mathbb{F}_2^n$, $u \in \mathbb{F}_2^{\frac{n}{2}}$, and $v \in \mathbb{F}_2^{\frac{n}{2}}$. We denote by $P_n(z)$ the polarization vector of z, and define the following recursive rule:

$$P_n(z) = \left[P_{n/2}(u \oplus v), P_{n/2}(v)\right],$$

where $P_1(z) = z$. This recursive approach constructs the polarization vector by combining the polarization of the XOR operation between the two halves and the polarization of the second half. To illustrate this new recursive approach, we will demonstrate an example with $n = 4$. Let $z^4 = [z_1, z_2, z_3, z_4]$. We set $u = [z_1, z_2], v = [z_3, z_4]$. Then,

$$P_4(z^4) = [P_2([z_1 \oplus z_3, z_2 \oplus z_4]), P_2([z_3, z_4])]$$

$$= [z_1 \oplus z_2 \oplus z_3 \oplus z_4, z_2 \oplus z_4, z_3 \oplus z_4, z_4],$$

as expected. This recursive formula can be use to calculate the polarization complexity.

B. Polarization Complexity

Lemma 1 (Polarization complexity time) Let $T(n)$ be the complexity time of polarization of a vector $z \in \mathbb{F}_2^n$. Then, $T(n) = O(n \log n)$, where big O notation is defined in Definition 2.

Proof: In each recursive step we perform the following:

- **Splitting**: The input vector of length n is split into two vectors of length $\frac{n}{2}$.
- **XOR Operations** The XOR operation is performed between the two halves.
- **Recursion** The process is then applied recursively to each of these halves.

Given this, the total number of operations performed can be expressed using the recurrence relation

$$T(n) = 2T(\frac{n}{2}) + O(n),$$

where $O(n)$ represent the linear work done in each step for the XOR operations between the halves. This recurrence relation is a classic example of the divide-and-conquer strategy. Note that

$$\begin{aligned} T(n) &= 2T\left(\frac{n}{2}\right) + O(n) \\ &= 2\left[2T\left(\frac{n}{4}\right) + O\left(\frac{n}{2}\right)\right] + O(n) \\ &= 4T\left(\frac{n}{4}\right) + O(n) + O\left(\frac{n}{4}\right), \end{aligned}$$

and in general, one can prove using induction, that:

$$\begin{aligned} T(n) &= 2^{\log n} T(1) + \sum_{i=0}^{\log n} O\left(\frac{n}{2^i}\right) \\ &= nT(1) + O(n \log n), \end{aligned}$$

and thus $T(n) = O(n \log n)$.■

C. Decoding compressed data

We now provide a detailed explanation of how the decoder operates. As previously mentioned, the decoder receives k bits corresponding to the unpredictable bits $w_{i_1}, \ldots, w_{i_k}$. The goal is to estimate the remaining $(n-k)$ bits using the following method:

$$\hat{w}_{i_j} = \arg\max_b P(w_{i_j} = b | W^{i_j - 1}).$$

To compute this argmax, the decoder must estimate the distribution of w_{i_j}. This process involves utilizing the unpredictable bits in a manner similar to how we used the frozen bits in channel coding. We will start with the case where $n = 2$. Let Z_1 and Z_2 be two independent random variables distributed as Ber(p_1) and Ber(p_2), respectively. Assume that $w_1 = z_1 \oplus z_2$. To calculate $\underset{b}{\text{argmax}} P(w_2 = b | w_1)$, the decoder must estimate $P(w_2|w_1)$. Define

$$b^+(p_1, p_2) := p_1(1-p_2) + p_2(1-p_1),$$

and note that:

$$\begin{aligned} P(w_1 = 1) &= P(z_1 = 1, z_2 = 0) + P(z_1 = 0, z_2 = 1) \\ &\overset{(a)}{=} P(z_1 = 1) P(z_2 = 0) + P(z_1 = 0) P(z_2 = 1) \\ &= p_1(1-p_2) + p_2(1-p_1) \\ &\overset{(b)}{=} b^+(p_1, p_2), \end{aligned}$$

where (a) stems from the assumption that Z_1, Z_2 are independent and (b) is due to the definition of b^+. Next, we observe that:

$$\begin{aligned} P(w_2 = 1 | w_1 = 0) &= P(z_2 = 1 | z_1 \oplus z_2 = 0) \\ &= \frac{P(z_2 = 1, z_1 \oplus z_2 = 0)}{p(z_1 \oplus z_2 = 0)} \\ &= \frac{p_1 p_2}{p_1 p_2 + (1-p_1)(1-p_2)}, \end{aligned}$$

and similarly:

$$P(w_2 = 1 | w_1 = 1) = \frac{p_1(1-p_2)}{p_1(1-p_2) + p_2(1-p_1)}.$$

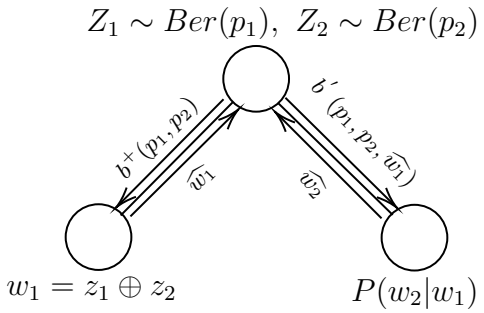


Fig. 6: Successive cancellation scheme for $n = 2$.

Thus, using $\hat{w_1}$, we can estimate the distribution $P(w_2|w_1)$ and determine $\hat{w_2}$ by taking the argmax. Let us denote this estimation of $P(w_2|w_1)$ with $b^{'}(p_1, p_2, \hat{u_1})$. The successive cancellation scheme for $n = 2$ is illustrated in Figure 6, where the process begins at the left child node to achieve $\hat{w_1}$. Using $\hat{w_1}$ we move to the right child to estimate $\hat{w_2}$. Then, $\hat{w_1}$ and $\hat{w_2}$ are propagated upwards to the parent node. Finally, given $\hat{w_1}, \hat{w_2}$ and the polarizing matrix P, one can reconstruct z_1, z_2 using

$$\hat{z}^n = \hat{w}^n \cdot P^{-1}, \tag{1}$$

where P is the polarizing matrix used to transform z_1, z_2 into w_1, w_2.

Given an arbitrary n, this process naturally generalizes in a recursive manner using a tree diagram, similar to our discussion of channel coding. Indeed, given a probabilities vector $\overline{p} = [p_1, ..., p_n]$, we divide the vector into 2 sub-vectors $\overline{p^1} = \left[p_1, ..., p_{\frac{n}{2}}\right]$, and $\overline{p^2} = \left[p_{\frac{n}{2}+1}, ..., p_n\right]$. Then we move to the left child, and calculate $b^+\left(\overline{p^1}, \overline{p^2}\right)$ to receive $\hat{w}_{\text{left}}$. Utilizing $\hat{w}_{\text{left}}$ we proceed to the right child and estimate $\hat{w}_{\text{right}}$ using $b^{'}\left(\overline{p}_1, \overline{p}_2, \hat{w}_{\text{left}}\right)$. Finally the parent node propagates upwards $[\hat{w}_{\text{left}} \oplus \hat{w}_{\text{right}}, \hat{w}_{\text{right}}]$ using the values estimated by each of the childs. Once the root gathers all the estimations from its successors using equation 1. We provide another example of successive cancellation for $n = 4$ in Figure 7.

Remark 1 We have seen an (n, k) code compression using polar codes. In this setup, errors may always occur due to errors in the estimation of the predictable bits using $\underset{b}{\text{argmax}} P\left(w_n = b|w^{n-1}\right)$. Assume dynamic coding is acceptable, meaning we allow the

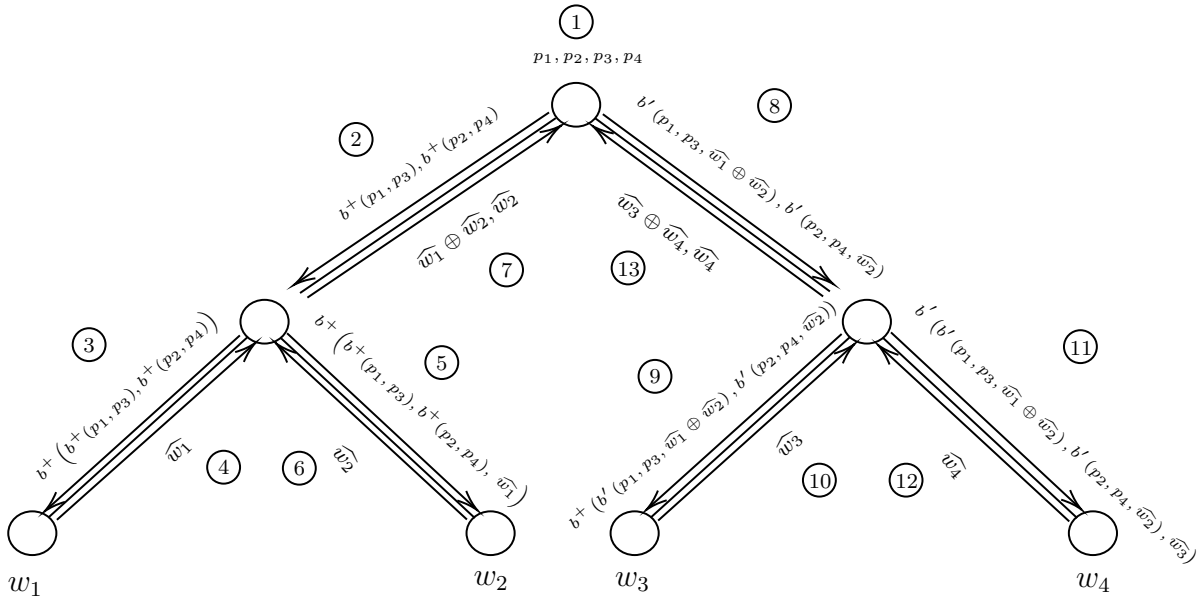


Fig. 7: Successive cancellation scheme for $n = 4$

increase in the number of transmitted bits while still maintaining the requirement that the expectation of the rate remains low. What modifications should be made to the encoder to ensure totally lossless transmission?

Answer: Since we assume the channel is not noisy, the encoder can accurately determine the values that will be received by the decoder. Therefore, for each predictable bit, the encoder can calculate the prediction that will be made by the decoder. If an error occurs, the encoder can transmit the corresponding bit to correct the decoder's erroneous estimation. This dynamic adjustment ensures that the transmission remains lossless while keeping the expected rate low.

IV. Theoretical Analysis

Throughout this theoretical discussion, consider:

- $P \in \mathbb{F}_2^{n \times n}$ - invertible matrix.
- $Z^n = (Z_1, Z_2, ..., Z_n) \sim Bern(p)^n$ - n size vector of i.i.d $Bern(p)$.
- $W^n \triangleq Z^n \cdot P$

- $W^i = (W_1, W_2, ..., W_i)$.
- We say that W_i is highly predictable given W^{i-1} if $H(W_i|W^{i-1}) \leq \tau$ for a small τ.
- $S = S_\tau = \{i \in [1, ..., n] | H(W_i|w^{i-1}) > \tau\}$ - the set of unpredictable elements.

We start with several properties regarding the set S. First, in the following lemma, we introduce an upper bound on the entropy of W^n.

Lemma 2 (Upper bound on $H(W^n)$) Let S be the set of unpredictable elements of W^n. Then we have that $H(W^n) \leq |S| + n \cdot \tau$.

Proof:

$$\begin{aligned} H(W^n) &\overset{(a)}{=} \sum_{i=1}^{n} H(W_i|W^{i-1}) \\ &\overset{(b)}{=} \sum_{i \in S} H(W_i|W^{i-1}) + \sum_{i \notin S} H(W_i|W^{i-1}) \\ &\overset{(c)}{\leq} |S| + \tau(n - |S|) \\ &\leq |S| + n \cdot \tau, \end{aligned}$$

where (a) follows from the chain rule, (b) follows from sum property, and (c) follows from the fact that $H(X|Y) \leq H(X)$ and $H_b(W_i) \leq 1$ (W_i is binary).

Moving forward, using Lemma 2, we provide in the following lemma a lower bound on the set S of unpredictable elements of W^n.

Lemma 3 (Size of S) Let $Z^n = (Z_1, Z_2, ..., Z_n) \sim Bern(p)^n$ and $P \in \mathbb{F}_2^{n \times n}$ invertible matrix. Further, let $W^n \triangleq Z^n \cdot P$ and S be the set of unpredictable elements of W^n. Then, the size of the set S is lower bounded by $|S| \geq n \cdot (H(p) - \tau)$.

Proof: Consider the following:

$$\begin{aligned} |S| + n \cdot \tau &\overset{(a)}{\geq} H(W^n) \\ |S| + n \cdot \tau &\overset{(b)}{\geq} H(Z^n) \\ |S| + n \cdot \tau &\overset{(c)}{\geq} n \cdot H_b(p), \end{aligned}$$

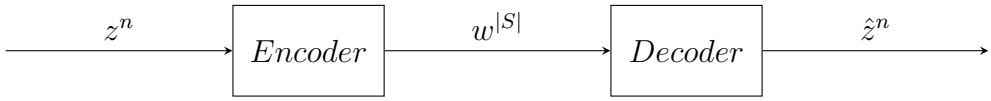


Fig. 8: Encoder-Decoder system.

where (a) follows from Lemma 2, (b) follows from the fact that P is invertible, and (c) follows from the fact that z^n is distributed i.i.d $Bern(p)$. Accordingly, we obtain that

$$|S| \geq n \cdot (H_b(p) - \tau).$$

In conclusion, Lemma 2 provides a lower bound on the size of the set S, which contains the indices of the unpredictable elements of the polarized vector W^n. This result is derived by using the chain rule of entropy and some properties of entropy to relate the entropy of W^n to the entropy of the original data symbols Z^n and the conditional entropy of each bit W_i given the previous bits W^{i-1}. This lemma provides a useful tool for estimating the size of S and understanding the behavior of polar codes.

To sum up and relate to the practical examples we have seen earlier, our process is illustrated in Figure 8, where z^n is an i.i.d vector $\sim Bern(p)^n$, $w^n \triangleq z^n \cdot P$, and S is equal to the set of unpredictable elements of w^n. The encoder receive z^n and outputs $w^{|S|}$ which consist of the w_i for which $i \in S$. The decoder receive $w^{|S|}$ to compute $\hat{z}^n$. Note that:

- We use $|S|$ bits over n bits message, and therefore the rate is $R = \frac{|S|}{n}$.
- At lecture 5 we have shown that $R \geq H(z^n)$.

The decoder operates according to the successive cancellation decompression (SCD) algorithm. The SCD algorithm works as follows:

- Input: $w^{|S|}$, $S \subseteq [1, ..., n]$, $P \in \mathbb{F}_2^{n \times n}$.
- Output: $\hat{z}^n \in \mathbb{F}_2^n$.
- Algorithm:

```
SCD(w^|S|, S, P)
    for i = 1 to n:
```

```
        if i ∈ S:
            ŵ_i = w_i
        else:
            ŵ_i = arg max_b {P_{W_i|W^{i-1}}(b|ŵ^{i-1})}
    return ẑ^n = ŵ^n · P^{-1}
```

Our decoder is taking the bits we sent $w^{|S|}$ and building new vector $\hat{w}^n$ of size n as follows. For indices $i \in S$ (that correspond to unpredictable bits) we set $\hat{w}_i = w_i$. Else, it means that $\hat{W}_i$ is predictable so we can predict him. After we got $\hat{w}^n$ we can compute $\hat{z}^n$ by the inverse matrix P, i.e., $\hat{z}^n = \hat{w}^n \cdot P^{-1}$

A. Theorem Error of source coding

In the following we introduce a theorem which provides a theoretical bound on the error rate of SCD algorithm when using polar codes for compression. We start with the following definition of an (ϵ, τ) polarizing:

Definition 1 (Polarized matrix & unpredictable columns) We say that an invertible matrix $P \in \mathbb{F}_2^{n \times n}$ is (ϵ, τ)-polarizing for $Bern(p)^n$ if for $W^n = Z^n \cdot P$ (where $Z^n \sim Bern(p)^n$) and

$$S = S_\tau = \{i \in [1, ..., n] \mid H(W_i|W^{i-1}) > \tau\}$$

we have $|S| \leq n \cdot (H(p) + \epsilon)$.

The theorem states that if the polarizing matrix P is (ϵ, τ) polarizing with unpredictable columns S, then the failure probability of the SCD is $\tau \cdot n$, where n is the length of the code. This means that the probability that the original data symbols z^n cannot be recovered exactly from the compressed representation $\hat{z}^n$ is less than or equal to $\tau \cdot n$.

The theorem also states that if P is $(\epsilon, \epsilon/n)$ polarizing, then the error rate of the SCD decoder is at most ϵ. This means that by choosing an appropriate polarizing matrix P, it is possible to control the error rate of the SCD decoder and achieve a desired level of accuracy in recovering the original data symbols from the compressed representation.

In summary, this theorem provides a theoretical bound on the error rate of successive cancellation decompression when using polar codes for compression and shows how the choice of polarizing matrix can affect the accuracy of data recovery.

Theorem 1 If P is (ϵ, τ) polarizing with unpredictable columns S then the SCD has failure probability of $\tau \cdot n$, i.e.

$$\mathbb{P}_r(z^n \neq \hat{z}^n) \leq \tau \cdot n,$$

where $\hat{z}^n = SCD\Big((z^n \cdot P)_s, P, S\Big)$. Accordingly, for P that is $(\epsilon, \frac{\epsilon}{n})$ polarizing, we have that

$$\mathbb{P}_{err} \leq \epsilon.$$

Thus, if P is $(\epsilon, \frac{\epsilon}{n})$ polarizing, the error of SCD decoder is at most ϵ.

Before we can prove this theorem we will need to prove the following lemma.

Lemma 4 Let X be random variable with $H(X) \leq \alpha$. Then:

1) There exists $x \in X$ such that $\mathbb{P}_r(X \neq x) \leq \alpha$.
2) Let (X, Y) be jointly distributed random variable with $H(X|Y) \leq \alpha$, and $A(y) = \arg\max_{x \in X}(\mathbb{P}(X = x|Y = y))$. Then $\mathbb{P}_r(X \neq A(y)) \leq \alpha$.

Proof (part 1): Let $p_i \triangleq \mathbb{P}_r(X = i)$, $x \triangleq \arg\max(p_i)$, and $p_x = 1 - \gamma$ s.t. $\mathbb{P}_r(X \neq x) = \gamma$.

Test case 1 ($\gamma \leq \frac{1}{2}$)

$$\begin{aligned}
\alpha &\geq H(X) \\
&= \sum_{i=1}^{|X|} p_i \log\left(\frac{1}{p_i}\right) \\
&\overset{(a)}{\geq} \sum_{i \notin x} p_i \log\left(\frac{1}{p_i}\right) \\
&\overset{(b)}{\geq} \sum_{i \notin x} p_i \log\left(\frac{1}{\sum_{j \notin X} p_j}\right) \\
&\overset{(c)}{\geq} \gamma \log\left(\frac{1}{\gamma}\right)
\end{aligned}$$

$$\geq \quad \gamma,$$

where (a) follows from subtracting a positive number from the sum of positive numbers, (b) follows from the inequality $p_i \leq \sum_{j \notin X} p_j$ where $i \notin X$, and (c) follows from the equality $P_r(X \neq x) = \gamma$.

Test case 2 ($\gamma \geq \frac{1}{2}$)

$$\begin{aligned}
\alpha &\geq H(x) \\
&= \sum_{i=1}^{|X|} p_i \log(\frac{1}{p_i}) \\
&\overset{(a)}{\geq} \sum_{i=1}^{|X|} p_i \log(\frac{1}{p_x}) \\
&\overset{(b)}{=} \log(\frac{1}{p_x}) \geq 1 \\
&\overset{(c)}{\geq} \mathbb{P}_r(X \neq x),
\end{aligned}$$

where (a) follows from the inequality $\forall i \in [1, |X|] p_x \geq p_i$, (b) follows from the quality $\log(\frac{1}{p_x})$ is a constant and $\sum_{i=1}^{|X|} p_i = 1$, and (c) follows from the quality $\sum_{i=1}^{|X|} p_i = 1$.

Proof (part 2):

$$\begin{aligned}
\mathbb{P}_r(X \neq A(Y)) &\overset{(a)}{=} \sum_y \mathbb{P}_r(X \neq A(Y), y) \\
&\overset{(b)}{=} \sum_y \mathbb{P}_r(y) \mathbb{P}_r(X \neq A(Y)|y) \\
&\overset{(c)}{\leq} \sum_y \mathbb{P}_r(y) H(X|Y = y) \\
&\overset{(d)}{=} H(X|Y) \\
&\leq \alpha,
\end{aligned}$$

where (a) follows from the law of total probability, (b) follows from the Lemma 2 in part (1), (c) follows from the chain rule, and (d) follows from conditional entropy law.

In summary, Lemma 2 provides two results for bounding probabilities of events involving random variables with bounded entropy. The first result states that if the entropy of a random variable X is less than or equal to α, then there exists a value x such that the probability that X is not equal to x is also less than or equal to α. The second result states that if X and Y are jointly distributed random variables with conditional entropy $H(X|Y)$ less than or equal to α, then the probability that X is not equal to the value that maximizes the conditional probability of X given Y is also less than or equal to α.

Proof (Theorem 1): Let $W^n \triangleq Z^n \cdot P$, and assume that for every $i \in S$ we set $\hat{W}_i = W_i$. For any $i \notin S$, by applying Lemma 2 part (2) with $X = W_i$, $Y = W^{i-1}$ and $\alpha = \tau$ we get that

$$\mathbb{P}_r(W_i \neq A_i(w^{i-1})) \leq \tau. \tag{2}$$

where $A_i(\cdot)$ is

$$A_i(w^{i-1}) = \arg\max_{w_i}(\mathbb{P}(W_i = w_i | W^{i-1} = w^{i-1})).$$

By union the bound, we get that

$$\sum_{i=1}^{n} \mathbb{P}_r(W_i \neq A_i(w^{i-1})) \leq \tau \cdot n.$$

We can note that the SCD algorithm used $A_i(\cdot)$ that is equal to the $\arg\max$. If $w^n \neq \hat{w}^n$ there exists i s.t. $\hat{w}_i \neq w_i$, which gives us $\mathbb{P}_r(w^n \neq \hat{w}^n) \leq \tau \cdot n$. Accordingly, we get

$$\mathbb{P}_r(z^n \neq \hat{z}^n (= SCD(z^n \cdot P)_s, P, S)) \leq \tau \cdot n.$$

Theorem 2 (Strong polarization) Fix $p \in (0, \frac{1}{2})$ and constant c. There exists a polynomial function n_0 such that for every $\epsilon > 0$, there exists $n = 2^t$ with $\frac{1}{\epsilon} \leq n \leq \frac{n_0}{\epsilon}$, and a set $E \subseteq \{1, \ldots, n\}$ with $|E| \leq \frac{\epsilon}{2} \cdot n$ such that for every $i \notin E$, the conditional entropy $H(W_i|W^{i-1})$ is either less then n^{-c}, or greater then $1 - n^{-c}$. Furthermore, if we let $S = \{i \in [n] | H(W_i|W^{i-1}) \geq n^{-c}\}$ then $|S| \leq (H(p) + \epsilon)n$ and the matrix P_n is $(\epsilon, 1/n^c)$ polarizing for $BER(p)^n$ with unpredictable columns S.

The strong polarization (Theorem 2) allows us to specify how close to zero the conditional entropy of the polarized bits should be.

In conclusion, we have presented the topic of compression using polar codes, which are a novel type of error-correcting codes that have great potential for communication and compression applications. We have explained the concept of polarization, which is the key idea behind polar codes, and how it separates the entropy of equally entropic bits into more entropic and less entropic bits by using a polarization matrix. We have also shown how polar compression operates by encoding only the less entropic bits and using the SCD algorithm to decode them with low failure probability. Further, we have provided some theoretical results and proofs to demonstrate the properties and performance of polar codes.

For more information or more detailed proofs on polar codes and compression, please refer to [1].

REFERENCES

[1] V. Guruswami, A. Rudra and M. Sudan, *LaTeX: Essential Coding Theory*, 2nd ed. Cambridge, MA: MIT Press, 2022.

V. Appendix

Definition 2 (Big O notation) Let $T(n)$ be a real valued sequence. We say that $T(n) = O(f(n))$ if there exist constants $c \in \mathbb{R}, c > 0$ and $n_0 \in \mathbb{N}$, such that for all $n \geq n_0$, $T(n)$ satisfies:

$$T(n) \leq c \cdot f(n).$$

In the context of complexity analysis, $T(n)$ can describe the running time of an algorithm and $f(n)$ is a function representing the growth rate.